Organizing machine learning data flows

You can use Splunk’s Machine Learning Toolkit (MLTK) and the Deep Learning Toolkit (DLTK) for Splunk to address security, operations, DevOps or business use cases. However, when setting these apps up, it might not always be clear how best to organize the data flow in Splunk Enterprise or Splunk Cloud.

You can use basic, intermediate or advanced patterns, depending on the complexity of your use case, which will help you improve your existing or future machine learning workflows with MLTK or DLTK.

Basic pattern: Directly fit and apply

The MLTK relies mostly on the classical machine learning paradigm of “fit and apply”, where usually there are two main moving parts.

- You can train a model with the | fit ... into model statement:
  - Either in an ad-hoc search, for example, for scoring your model
  - Or in a scheduled search, for example, for retraining your model in a continuous way.
- Then you can use another search that uses | apply model to run it:
  - Either ad-hoc on new data
  - Or in a scheduled search that generates a report or an alert based on your model results.

This basic pattern works fine for many simple use cases and you can create productive machine learning pipelines with MLTK. While this pattern is great for quick and ad-hoc tasks, you will probably need to move on to another stage when your data flows get more complicated.

Intermediate pattern: Summary indexing

This pattern could be a good option for you when your model depends on multiple data sources and involves heavier SPL lifting. You might find you need a pattern like this when you need more preprocessing steps like data transformations for cleaning, merging, munging, or feature engineering in general.

Using Summary Indexing you can write search results as key=value pairs into an arbitrarily defined event index, or into

The information provided in Splunk Lantern is intended for informational and educational purposes only. All information is provided in good faith, however, Splunk disclaims any and all representations and warranties, express and implied, regarding the information provided, including without limitation any warranties and representations regarding the completeness, adequacy or accuracy of the information. You agree to take full responsibility for the results arising from the use of the information provided.
a **metrics** index for purely numerical time series data.

This helps offload ongoing calculations of features and statistics into a separate index. A separate index is often much faster to query instead of searches which run repeated heavy computations on large sets of historical data. **Data models** can also be used to accelerate the process.

For MLTK, you will need to run your | **fit** on a summary index or a data model instead of the raw data. This can significantly speed up overall processing time, especially when there is more more heavy lifting required on the SPL side.

As the | **apply** is often computationally not as demanding as the | **fit**, you can either apply all transformations on the new raw indexed data then or - with a certain time lag - on the summary indexed data.

### Advanced pattern: Enrichment with feedback

This pattern uses the human-in-the-loop type of machine learning systems that allow users to add further data into the training pipeline. This data could either be additional features in terms of synthetically generated data points, or changing parameters - for example, to adjust the sensitivity of an anomaly detection model.

As long as the feedback can change over time but is related to some unique identifier, then Splunk’s **KVStore** is a good fit for that requirement and allows for further enrichment of the training data, for example by incorporating user generated labels for a classifier model. The **DGA App for Splunk** shows an example for that pattern that could help you get started quickly.

You could also combine the KVStore with summary indexing to keep track of all changes over time. This method can be useful for auditing purposes, as well as to create another data layer which can be utilized for further supervised
learning tasks, as it contains labels generated for the given cases.

Next steps

The content in this guide comes from a previously published blog, one of the thousands of Splunk resources available to help users succeed. In addition, these Splunk resources might help you understand and implement this use case:

- .Conf Talk: SEC1374 - Augment your security monitoring use cases with Splunk's Machine Learning Toolkit
- Blog: MLOps - Logs, metrics and traces to improve your machine learning systems